Refine Search

Search Results -

Terms	Documents
L13 and L1	1

Database: US Pre-Grant Publication Full-Text Database; US Patents Full-Text Database; US OCR Full-Text Database; EPO Abstracts Database; JPO Abstracts Database; Derwent World Patents Index; IBM Technical Disclosure Bulletins

Search: L14

Refine Search | Recall Text | Clear | Interrupt

Search History

DATE: Thursday, May 26, 2005 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L14	L13 and l1	1	L14
L13	l2 same bus	8	L13
L12	L2 and l3	1	L12
L11	L10 and l1	4	L11
L10	L8 same test$	4	L10
L9	L8 and l1	52	L9
L8	L6 same bus	52	L8
L7	L2 same l3	1	L7
L6	L1 same l3	53	L6
L5	L1 same l2	0	L5
L4	L3 same l2 same l1	0	L4
L3	motherboard	12760	L3
L2	traffic adj1 generator	139	L2
L1	arbiter	9784	L1

Hit List

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Search Results - Record(s) 1 through 1 of 1 returned.

☐ 1. Document ID: US 6877053 B2

L14: Entry 1 of 1 File: USPT Apr 5, 2005

DOCUMENT-IDENTIFIER: US 6877053 B2
TITLE: High performance communication architecture for circuit designs using probabilistic allocation of resources

Brief Summary Text (20):
Since buses are often shared by several masters, bus architectures come with mechanisms or protocols to manage access to the bus. These mechanisms and protocols are implemented in centralized (or distributed) bus arbiters. Approaches for managing access to a shared resource include round-robin access, priority based selection, and time division multiplexing. It should be noted that, while the discussions may emphasize buses as a shared resource, the disclosed techniques can be applied to any shared resource.

Brief Summary Text (26):
The shared system bus with a static priority based arbitration protocol is one of the commonly used bus architectures [1]. Such a bus (as shown in FIG. 1) 1.30 is a set of address, data, and control lines that are shared by a set of masters 1.21-1.24 that contend among themselves for access to one or more slaves 1.51-1.54. A centralized bus arbiter 1.10 periodically examines accumulated requests from the various master interfaces, and grants bus access to the master that holds the highest priority among the contending masters 1.21-1.24. The bus also supports a burst mode of data transfer, where the master negotiates with the arbiter to send or receive multiple words of data over the bus without incurring the overhead of handshaking for each word. The maximum size of this transfer is a parameter of the architecture, and is defined along with the various component priorities as shown in FIG. 1. Other parameters that need to be chosen for this architecture include the width of the bus (bytes per word), its frequency of operation, and the address space associated with each slave interface. Some example parameters are shown in 1.40. Several flavors of basic priority based arbitration may be created by combining the basic protocol with one or more enhancements. These include support for pre-emptive transactions, multi-threaded transactions, dynamic bus splitting. etc.

Brief Summary Text (31):
To avoid wasting cycles, arbitration is pipelined with word transfers, as shown in FIG. 2(b). For example, while D1 (data from M.sub.1) is begin transferred, the next slot's grant is computed as G-2, since M.sub.2 has data to send. However, for slots reserved by M.sub.4, the arbiter detects that M.sub.4 has no requests, and therefore assigns them to M.sub.1 and M.sub.2, in round-robin sequence. The 6.sup.th slot is unutilized as there are no pending requests.

Brief Summary Text (41):
In this example, the static priority based architecture described in Section I.D.2.

(a) is discussed. The manner in which it allocates the bandwidth of the bus to the various components that are connected to it is illustrated. For this experiment, the system shown in FIG. 3(a) is considered. The system consists of a single bus 3.10 with four masters 3.21-3.24, which contend with each other for access to a shared memory 3.30. For this experiment, the bus is always busy, i.e., frequent requests for access to the bus ensure that at any given instant, at least one request from the set of masters is awaiting service. To enforce the bus protocol described in Section I.D.2(a), the bus masters are assigned unique priority values. A priority level 4 is assigned to the component of highest priority, 3 to the next highest, and so on. The fraction of the bandwidth assigned to each component under the given priority assignment was measured over a long simulation trace, in which requests for access to the bus were modeled using stochastic on-chip communication traffic generators. The simulation was repeated for every possible priority assignment, to generate the results shown in FIG. 3(b).

Detailed Description Text (60):
The aim of the first set of experiments is to test the ability of LOTTERYBUS architecture to provide proportional bandwidth allocation for widely varying characteristics of the communication traffic on the bus. Several different types of traffic were generated. ranging from regular periodic traffic with large inter-request intervals, to random bursty traffic with high frequency of requests, by setting appropriate parameter values in the traffic generators. For each class of traffic, the system was simulated for a long sequence of input stimuli, and the bandwidth allocated to each SoC component was measured. The results are shown in FIG. 13. The x-axis depicts the nine different classes of communication traffic that were considered, the y-axis depicts the fraction of the total bus bandwidth allocated to various components, as well as the fraction that remains unutilized.

Detailed Description Text (65):
In the second set of experiments, the performance of the LOTTERYBUS architecture, that embodies the disclosed techniques, and the TDMA based communication architecture is compared. Again, the traffic generators are used to create a set of classes of on-chip communication traffic. The performance metric that is considered for this experiment is the average latency (in clock cycles per word) of transferring a word across the bus. This includes the waiting time due to bus conflicts, the overhead of arbitration, and the time spent in executing the data transfer. In this experiment, the point of interest is in assessing and comparing the ability of the TDMA architecture and the LOTTERYBUS architecture in being able to provide low communication latencies to high priority traffic. Thus the traffic classes chosen included ones where one component makes frequent high priority burst requests. To study how the two architectures compare in being able to support such communications, the bursty component was assigned the most slots in the TDMA architecture, and the largest number of tickets in the LOTTERYBUS architecture. In fact. both the slot reservations in the TDMA wheel and the tickets were assigned in the ratio 1:2:3:4.

Other Reference Publication (1):
"Priority arbiters" by Bystrov, A.; Kinniment, D.J.; Yakovlev, A. (abstract only).*